

needfile, ndflist, and ndfcheck Commands

product needfile - release v5_0

Introduction

The **needfile** interface was developed in order to:

- Avoid tying up tape drives for long periods of time.
- Avoid excessive mounts of the same user tape.
- Allow multiple jobs to reference files from the same tape.
- Provide consistent error recovery for serial media..
- Allow accessing the contents of tapes from nodes without suitable tape drives.

Changes from previous versions

needfile now uses a Linux Terabyte disk server to store its data, and anonymous FTP for file access. The **needfile** interface of the obsolete **fmss** product is no longer available.

The new **-of** option can be used to specify the destination local file, without using the **FERMINT_DPOOL_DIR** environment variable.

FERMINT_DPOOL_DIR is not declared automatically when the **needfile** product is set up.

Implementation

needfile is implemented as a client of the servers that manage the tapes and files. There is a **needfile** prestager, that controls the copying of tapes into the needfile cache. When a **needfile** command is issued, the following steps take place.

Step 1:

If you issue a **needfile** command interactively, and the tape is not prestaged, you will be asked whether to prestage the tape. You can use the **-stage** option to explicitly request prestaging.

Step 2:

The **needfile** prestager informs the operator to mount the tape on an available drive, and copies its contents onto disk. You can request files after they have been copied.

Step 3:

The **needfile** command is executed, and contacts the **needfile** server. The server replies with information needed to locate the file.

Step 4:

The **needfile** command processor contacts the Linux disk cache server and retrieves the requested file.

ndflist is a utility to list and describe currently available and active **needfile** tapes.

ndfcheck is a utility to query the status of the **needfile** servers.

Before using needfile

The **needfile** product must be set up to use the commands described in this document.

Some options for the **needfile** command require that you define the **FERMINT_DPOOL_DIR** environment variable. The value of **FERMINT_DPOOL_DIR** must be an absolute path to a directory where files retrieved by **needfile** are to be stored.

Format of the needfile command

needfile **-lb** *file* **-vs***n* *vs**n* **-file** *nnn* [options]

needfile **-of** *path* **-vs***n* *vs**n* **-file** *nnn* [options]

needfile **-clear** *file*

needfile **-clearall**

needfile **-clobber**

needfile **-free** **-vs***n* *vs**n* [**-u***fort*]

needfile **-query** **-vs***n* *vs**n* $\left[\begin{array}{l} \text{-- file } nnn \\ \text{-- name } name \end{array} \right]$ [**-u***fort*]

needfile **-stage** **-vs***n* *vs**n* [**-email** *email_address*] [options]

needfile **-version**

needfile **-wait** *local-file*

Description

needfile allows UNIX batch jobs and interactive users to access the contents of data tapes. The tapes are read by a tape server, and the contents are stored by a data server and copied to the requesting UNIX node.

Functions

Exactly one of the following functions must be specified, unless the **-name** flag is used.

-lb *file*..... Access the retrieved data directly as the file name specified. The file name specified is relative to the directory specified in the **\$FERMINT_DPOOL_DIR** environment

variable. If **FERMINT_DPOOL_DIR** is not defined it will default to the current directory.

- of path** Access the retrieved data directly as the file name specified. The file name specified is a complete relative or absolute path.
- clear file**..... Delete a local file (within the **FERMINT_DPOOL_DIR** directory) created by a previous **needfile** command. If there is a pending request (**-nw** specified) for the file, it will be canceled.
- clearall**..... Delete all local files (within the **FERMINT_DPOOL_DIR** directory) created by previous **needfile** commands, and cancel all pending requests.
- clobber** Same as **-clearall**, but acts quietly and always returns 0.
- free**..... Indicate that the tape is no longer needed, or has been modified. Any future **needfile** requests for the tape will force it to be restaged. Any CLUBS batch jobs, currently in the job queue will probably fail.

Note that the **-ufort** flag may be required for proper execution of the **-free** function. See the usage notes.
- query** Report immediately the status of the requested volume/file combination .

Note that the **-ufort** parameter may be required for proper execution of the **-query** function. See the usage notes.
- stage**..... Stages the requested tape. This option is intended for interactive use.
- version**..... Display the version of the **needfile** program.
- wait file**..... Wait for completion of a previous **needfile** command, that had **-nw** specified.

Options

Options shown in certain forms of the **needfile** command are mandatory or the only ones allowed.

- block bbb** Physical tape block size when the unlabeled tape contains fixed length blocked records. The maximum value is 64000. The **-lrecl** option must also be specified, and the block size must be an exact multiple of **-lrecl**. If the **-block** parameter is omitted, and **-lrecl** is specified, the default is the value of **-lrecl**.
- email address** ... User to notify when a stage completes. This is valid only with the **-stage** function.
- file nnn** Indicates that the desired data on the tape is from the physical file indicated.
- files nnn** Prestaging of data from the requested tape will stop after *nnn* files have been read. The default value is 9999. The new **-range** option can be used instead to provide better control.
- keep** If a tape error is encountered, and some data has been read, keep the partially read file and do not retry. The default is to retry, and completely discard any file that is not totally readable.

- lrecl** *rrr* Data record size, when the unlabeled tape contains fixed length blocked records. The maximum value is 64000. The **-lrecl** parameter is meaningful only when **-ufort** is also specified.
- name** *name* Indicates that the desired data on the tape is from the file with the given label name. If no function is specified, **needfile** will default to **-lb** with the specified file name.
- Note: all letters in the specified name will be converted to upper case. The default **-lb** parameter used the specified label name after this translation.
- nw** Do not wait for needfile completion. A later **needfile** command specifying only **-wait** with the same file name can be used when access to the data is required. This option is valid only with the **-lb** function. The **FERMINT_DPOOL_DIR** environment variable must be defined to use this option.
- range** *range* Control which files are staged. The *range* argument consists of one or more file ranges separated by commas. No spaces are allowed. The allowed forms for the individual ranges are:
- | | |
|----------------|---|
| <i>mmm-nnn</i> | files <i>mmm</i> through <i>nnn</i> |
| <i>mmm-</i> | file <i>mmm</i> though end of tape |
| <i>-nnn</i> | beginning of tape through file <i>nnn</i> |
| <i>nnn</i> | file <i>nnn</i> |
- For example **-range -3,5,8-10,15-** will stage files 1, 2, 3, 5, 8, 9, 10, 15 and the rest of the tape past file 15. The specifier for the **-range** option should be limited to 80 characters.
- tape** *type* Specifies the type of the user's tape volume. The acceptable values are:
- 8MM** User's volume is an unlabeled 8 millimeter cassette.
- 8MMA** User's volume is an 8 millimeter cassette with ANSI labels for each file.
- 8MMD** User's volume is an unlabeled dual density 8 millimeter cassette.
- 8MMDA** User's volume is an 8 millimeter dual density cassette with ANSI labels.
- The default type is **8MMD**. The *type* parameter is case insensitive.
- ufort** When copying a tape, insert 4 bytes containing the record size before and after each record read. This produces a files that conforms to requirements of **FORTRAN** (AIX version) unformatted IO, and RBIO. See the usage notes for further details.
- verbose** Produce additional output showing the progress of the **needfile** command.
- vsn** *vsn* Volume serial name of the tape to be accessed. The tape *vsn* is case insensitive, and may consist of up to 6 letters and digits. Special characters are not allowed.

Usage notes for needfile

- I) Tape formats supported are:
 - 1) Unlabeled tapes containing multiple files followed by an empty file. The **-lrecl** and **-block** parameters may be used to specify blocking.

- 2) ANSI and VMS standard labeled tapes containing multiple files. The blocking of individual files is determined from information in the tape labels. The **-name** option may be used to request an individual file by its label name.
- II) The **-ufort** flag is a significant qualifier , and must be specified with the **-free** or the **-query** function to refer to a tape staged with **-ufort**. The same tape may be used both with and without **-ufort** specified.
- III) **needfile** returns the following result codes.
 - 0.. Operation successful.
 - 1.. ERROR in needfile command, or file not found for the **-query** option.
- IV) **needfile** will make local copies only of tape files that have already been prestaged.
If a **needfile** command is entered interactively and the file is not available, you will be given the option to prestage your tape.
- V) Because of space constraints, **needfile** will purge tapes as needed. Least recently used tapes and tapes owned by users with the largest data volume will be purged first. Because this purging is done transparently, there is no quota notification method. Currently, each group is allowed up to 100 gigabytes of files, before its oldest files are purged. If a purged tape is re-requested, the data will be restaged from the original tape.
- VI) There is no way to cancel a **needfile** stage request, once it has been issued. In case of emergency, the operators and the system staff can prevent a stage from continuing.

Examples

Note: sample replies from needfile are shown in italics

- I) The following **needfile** command can be used to access file 3 from the labeled 8mm cassette MYTAPE. The file will be accessible as \$FERMINT_DPOOL_DIR/file3:

```
needfile -vsn mytape -tape 8MMA -file 3 -lb file3
```

*file ***/file3 was loaded*
- II) The following **needfile** command can be used to access file 3 from the labeled 8mm cassette MYTAPE. The file will be accessible as file3 in the current directory:

```
needfile -vsn mytape -tape 8MMA -file 3 -of file3
```

*file ***/file3 was loaded*
- III) The following **needfile** command will retrieve the file, whose label name is file.alpha from tape MYTAPE. By default the file will be stored as \$FERMINT_DPOOL_DIR/FILE.ALPHA:

```
needfile -vsn MYTAPE -tape 8mma -name file.alpha
```
- IV) The following **needfile** commands can be used to retrieve file 5 of MYTAPE without waiting for the retrieval to complete, to make the job take less time. Later on when the file is needed, the job will wait, if necessary, for the file to be ready.

```
needfile -vsn mytape -tape 8mma -file 5 -nw -lb file05
(other commands)
needfile -wait file05
```

- V) The files from tape MYTAPE are no longer needed. The following **needfile** command will remove its contents from the data server,

```
needfile -vsn MYTAPE -free
```

VSN MYTAPE was freed

- VI) The following statement will cause the 8mm cassette MYTAPE to be staged, with any messages sent to “myself@fnal”..

```
needfile -stage -vsn mytape -email myself@fnal.gov
```

*A mail message will be sent to myself@fnal when done
request (interactive.fnclub.*****.******) will be prestaged*

- II) You can check on the progress of the staging request by using the **ndflist** command or by issuing a needfile query:

```
needfile -query -vsn mytape
```

The **ndflist** command has the capability to list multiple tapes, and describe the individual files of a tape.

The **needfile** query command can produce several types of message, depending on the progress of the prestage operation, Note that there is no way to differentiate between a prestage request that has not started, and the complete absence of a prestage request.

Tape vsn MYTAPE is not prestaged (prestage not started)

*Tape MYTAPE has ** files. It is prestaging since *****
Total size of this VSN is ** MB*

*Tape MYTAPE contains ** files and was prestaged in ****
Total size of this VSN is ** MB*

Format of the ndflist command

ndflist [-i] [-p prefix]

ndflist [-i] [-u username]

ndflist [-i] [vsn1 [vsn2 ...]]

Description

ndflist is a utility to list and describe currently available and active **needfile** tapes.

The following options apply to the **ndflist** command:

- i** Provide information about the individual files on the tapes listed. This **ndflist** program may require a considerable time to gather the information.

-p *prefix*..... List tapes (for all users) with vsns starting with the requested prefix.

-u *username* List tapes prestaged by the specified user.

vsnl [*vsnl2*...]..... List the specified tapes.

If neither **-p** nor **-u**, nor a vsn list is specified, **ndflist** will list tapes originally staged by the requesting user.

The ndfcheck command

ndfcheck

ndfcheck is a utility to query the status of the **needfile** servers, that takes no arguments. It will print a message and will return one of the following result codes.

- 0 **needfile** server is fully operational.
- 1 **needfile** server is suspended
- 2 **needfile** server is not operational, or cannot be contacted.